UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P O Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/769,594	01/30/2004	Ahmed K. Ezzat	200315891-1	8589

22879   7590   06/27/2008
HEWLETT PACKARD COMPANY
P O BOX 272400, 3404 E. HARMONY ROAD
INTELLECTUAL PROPERTY ADMINISTRATION
FORT COLLINS, CO 80527-2400

EXAMINER
GYORFI, THOMAS A

ART UNIT	PAPER NUMBER
2135	

NOTIFICATION DATE	DELIVERY MODE
06/27/2008	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

JERRY.SHORMA@HP.COM
mkraft@hp.com
ipa.mail@hp.com

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Application Number: 10/769,594
Filing Date: January 30, 2004
Appellant(s): EZZAT, AHMED K.

_____
John P. Wagner
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 4/18/08 appealing from the Office action mailed 9/27/07.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

6125447   GONG   9-2000

In order to establish an argument from inherency regarding the limitation "portions of code that are executed at the same privilege level", Examiner has enclosed herein:

Lindholm, T. and Yellin, F. "The Java™ Virtual Machine Specification, Second Edition" Excerpt from

Holub, Allen. Excerpt from "Programming Java threads in the real world, Part 1, Page 5 of 5"

Originally Published at JavaWorld.com on Sept. 1, 1998 (7 pages)

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claims 1-25 are rejected under 35 U.S.C. 102(b) as being anticipated by Gong (U.S. Patent 6,125,447).

Regarding claims 1, 12, and 17:

Gong discloses a method, computer system and computer readable medium for providing flexible protection by decoupling protection from privilege, comprising: enabling receipt of information describing two or more types of protection (col. 8 line 40 – col. 9, line 37); enabling receipt of information describing a relationship between said two or more types of protection and portions of code that are executed in a same privilege level of the computer system, where said relationship is not required to be linear (Ibid; and col. 9, lines 40-53); and enabling the association of said information describing two or more types of protection and said information describing said relationship with said portions of code (Ibid, and col. 10, lines 57-62) wherein a first portion of code allowing a second portion of code to access the first portion of code does not depend on the second portion of code allowing the first portion of code to access the second portion of code (col. 12, line 40 – col. 13, line 10). Per claim 12, Gong further discloses a memory unit and processor (col. 4, lines 25-45).

Regarding claims 2, 13, and 18:

Gong further discloses wherein said relationship is user-definable (col. 8, lines 45-63, noting that the ability for a user to set permissions on at least one's home directory and the contents therein was known in the art).

Regarding claims 3, 14, and 19:

Gong further discloses wherein said portions of code are domains and each of said types of

protections is defined in part by at least one or more domain attributes (col. 9, lines 40-55).

Regarding claims 4 and 20:

Gong further discloses wherein said one or more domain attributes includes a domain identifier

that specifies a unique value for a particular domain (col. 9, lines 5-20).

Regarding claims 5 and 21:

Gong further discloses wherein said one or more domain attributes includes a Private Key that

specifies a unique value that a particular domain must use for protecting each user that concurrently

uses a particular domain (col. 9, lines 5-37).

Regarding claims 6 and 22:

Gong further discloses wherein said one or more domain attributes includes a SharedCode

Key that specifies a value that a particular domain must use to access code associated with another

domain (col. 9, lines 25-37).

Regarding claims 7 and 23:

Gong further discloses wherein said one or more domain attributes includes a SharedData Key

that specifies a value that a particular domain must use to access data associated with another

domain (Ibid).

Regarding claims 8 and 24:

Gong further discloses wherein said one or more domain attributes includes a AllowOthers Key that specifies a value that a particular domain must use to access code associated with another domain in conjunction with said particular domain performing cross-domain switching to said other domain (col. 9, lines 25-37; col. 10, lines 27-40).

Regarding claims 9 and 25:

Gong further discloses wherein said one or more domain attributes includes a AccessOthers Key that specifies a value that a particular domain must use to request access of code associated with a particular domain on behalf of another domain (col. 9, lines 25-37; col. 10, lines 1-17).

Regarding claims 10 and 15:

Gong discloses a method and computer system for providing flexible protection by decoupling protection from privilege, comprising: detecting a request from a first portion of code to access a second portion of code, wherein said first and second portions of code are executed in a same privilege level of said computer system (col. 9, lines 54-67; col. 11, lines 40-65); determining whether said first portion of code is allowed to access said second portion of code based on information describing two or more types of protection and also based on information describing a relationship between said two or more types of protection, wherein said relationship is not required to be linear (col. 8, line 40 – col. 9, line 37); if said relationship specifies that said first portion of code may access said second portion of code, then allowing said first portion of code to access said second portion of

code (col. 12, lines 54-67); else not allowing said first portion of code to access said second portion of code (Ibid). Per claim 15, Gong further discloses a memory unit and processor (col. 4, lines 25-45).

Regarding claims 11 and 16:

Gong further discloses wherein said information describing said two or more types of protection and said information describing said relationships are associated with said portions of code and wherein the method further comprises retrieving said information describing said two or more types of protection and said information describing said relationships (col. 12, lines 10-40).

(10) Response to Argument

First, with respect to the claim objections of claims 22-25, Examiner was well aware that the claim numbering was corrected in the preliminary amendment filed 2/27/04 (see the Final Office Action of 9/27/07, page 3, paragraph #3); Examiner was simply reacting to the amendment of 7/19/07, wherein Appellant apparently disregarded said preliminary amendment and identified claims 22-25 on page 6 of said amendment as claims 28, 29, 22, and 23, respectively. Accordingly, the objections were levied merely to caution the Appellant to avoid making such errors in future correspondence with the Office.

With respect to Appellant's arguments against the rejections of the claims in view of the Gong reference, Examiner believes it would be helpful to analyze the claim language in detail to understand exactly what is being claimed in the instant invention. Exemplary claim 1 reads as follows:

1.   A method of providing flexible protection in a computer system by decoupling protection from privilege, the method comprising:

        enabling receipt of information describing two or more types of protection;

> enabling receipt of information describing a relationship between said two or more types of
> protection and portions of code that are executed in a same privilege level of the computer system,
> wherein said relationship is not required to be linear; and
>
> enabling the association of said information describing two or more types of protection and said
> information describing said relationship with said portions of code, wherein a first portion of code allowing
> a second portion of code to access the first portion of code does not depend on the second portion of
> code allowing the first portion of code to access the second portion of code.

The Gong invention is primarily directed toward a method of providing flexible protection in a computer system. It does this primarily through the use of "protection domains", which are defined *inter alia* as "a set of permissions granted to one or more principals" (col. 8, lines 42-43). Gong discloses an exemplary embodiment wherein there exists at least two distinct protection domains (domains I and J, see Figure 6; cf. col. 11, lines 55-64), each with their own set of permissions; this clearly satisfies the claim limitation "enabling receipt of information describing two or more types of protection".

The Gong invention further comprises a domain mapper component (element 248 of Figure 2) whose purpose is to receive the information in order to combine the permission(s) for one or more resources with a code identifier (in essence, the name of a portion of code: col. 3, lines 20-50; col. 9, lines 25-35) to which the permission is applicable: see col. 9, lines 40-53. The domain mapper, whose functionality is illustrated as Figure 4, creates a protection domain object that is used by the system to determine when a given portion of code may access a particular resource: see col. 12, lines 25-40. This satisfies at least the portion of the limitation "enabling receipt of information describing a relationship between said two or more types of protection and portions of code…of the computer system." Additionally, it is observed that Gong does not disclose whether or not the

relationship between the types of protection and the portions of code to be governed by the disclosed

system are "linear" in nature; indeed, the term "linear" does not appear anywhere within Gong's

disclosure. However, the claim limitation states "wherein said relationship **is not required to be**

linear"; accordingly, even assuming *arguendo* that it could be established that the relationship

between the protection domains and the portions of code were linear in the disclosed example, this

would still be acceptable as the claim does not actually forbid a linear relationship. Gong would have

had to teach that the only relationship possible for the domain mapper to establish would be a linear

one in order to teach away from the instant invention, for which Examiner maintains that no such

disclosure exists within the Gong reference.

With respect to Appellant's argument that "Appellant does not understand Gong to teach,

describe, or suggest 'portions of code...executed in a same privilege level of said computer system,'

as recited by independent claims 1, 10, 12, 15, and 17." (e.g. page 16 of the Appeal Brief, last

paragraph), Examiner respectfully submits that this feature is inherent to the Gong invention. In

analyzing this argument Examiner notes that the specification rather helpfully defines what one of

ordinary skill in the art would understand the claim term as, on page 1, paragraph 0003:

> [0003] In computer terminology, "privilege" determines what actions code is
> allowed to perform on information in a computer system. In most operating
> systems, the actions that code is allowed to perform is determined by what the
> privilege level the code executes or resides in. For example, usually code
> executes in one of two privilege levels known as kernel and user. Further, code
> that resides in the kernel has an unlimited amount of power to perform actions, such
> as accessing information any where in the computer system. In contrast, code that
> resides in the user level has a limited amount of power to perform actions. For
> example, code in the user level can not access directly code or data in the kernel.

Although Gong provides enough information in the disclosure to establish that the pertinent

portions of code are all operating at the same (albeit unstated) privilege level. To wit: the preferred

embodiment of the Gong invention exists as software running within a Java virtual machine (col. 6,

lines 55-62; col. 11, lines 40-55), preferably running as one or more threads (Ibid). Those of ordinary

skill in the art would recognize that a Java virtual machine is a layer of software designed to run

programs written in the Java language, which in theory allows for a single program to be run on any

and all manner of disparate and otherwise incompatible computing platforms (see sections 1.1 and

1.2 of the enclosed "Java Virtual Machine Specification reference", incorporated by reference into

Gong at col. 6, lines 60-62). In practice, there are subtle distinctions between virtual machines on

different platforms that a developer must account for in writing Java software. Relevant to the claim

limitation regarding privilege levels, the Holub reference teaches that, although Java threads may be

implemented in one privilege level on one platform (e.g. Kernel mode threads on a Windows NT

based virtual machine: page 3, first paragraph) while implemented in a different privilege level on

another platform (user-mode lightweight processes [LWP] on a Solaris based virtual machine: page 3,

second and third paragraphs), nevertheless a single virtual machine will be consistent: all threads will

be implemented the same way on a specific given virtual machine (see page 4, 2nd paragraph).

Given one Java virtual machine, one could have some threads operate at one privilege level while

other threads may operate at another privilege level, although one of ordinary skill in the art would be

wise to ensure that one's code would run correctly regardless of whichever privilege mode it may end

up running in (see the enumerated list on page 4 of Holub). Thus, Examiner maintains that all the

pertinent portions of code disclosed by Gong would inherently be running at the same privilege level,

as would be understood by one of ordinary skill in the art in light of the instant specification; and

therefore, the second limitation of the independent claims is recited by the prior art in its entirety.

The third and final limitation of the independent claims is substantially similar to the second, in that having received the information regarding the relationship between the protection domains and the portions of code (col. 9, lines 45-53), the domain mapper object actively associates this information by creating one or more protection domain objects (col. 9, line 54 through at least col. 10, line 40). This leads to the limitation that Examiner surmises is the source of Appellant's confusion regarding the prior art, "wherein a first portion of code allowing a second portion of code to access the first portion of code does not depend on the second portion of code allowing the first portion of code to access the second portion of code." A careful inspection of the claim language indicates that this limitation is intended to impose a restriction as to which portions of code may access **_each other_**; it says nothing about how two portions of code may or may not interact in order to access **_a common resource, separate and distinct from either portion of code_**. As Appellant argues in the Appeal Brief, page 12:

> Therefore, Appellant understands Gong to teach that portions of code (a.x, b.y, c.z) attempting to access data (/tmp/) are dependent on each other for permission to access the data.

And again on page 16:

> For at least this reason, Appellant understands Gong to teach that Gong's portion of code a depends on code b and c to access /tmp/temporary in contrast to "wherein a first portion of code allowing a second portion of code to access the first portion of code does not depend on the second portion of code allowing the first portion of code to access the second portion of code," as recited by independent claims 1, 12, and 17.

In contrast to Appellant's argument, Gong discloses at least three Java objects [portions of code], designated a, b, and c, each comprising at least one method (a.x, b.y, and c.z respectively: col. 11, lines 40-54); for purposes of this discussion, Examiner focuses on a.x and b.y, particularly as both b.y and c.z belong to the same protection group (Figure 6) and c.z is subsequently removed from the stack soon enough (Ibid, at lines 50-54). The ability for one portion of code to invoke another portion of code logically implies that said one portion of code may access said another

portion of code; and while a.x is explicitly disclosed as being able to invoke b.y (Ibid, at lines 48-49),

no corresponding disclosure exists to state definitively whether b.y is capable of invoking a.x (or any

method belonging to object a). All one knows for certain from the Gong disclosure is that a.x may

access b.y; in fact, it must be able to access it in order for the invention to function as disclosed. As a

result, it is clearly seen that b.y allowing a.x to access b.y does not depend on a.x allowing b.y to

access a.x[1], and this is *exactly* what is recited in the claims. Appellant's arguments regarding the

alleged lack of this limitation in the Gong disclosure are thus seen to be moot because the issue that

Appellant is actually arguing in the Appeal Brief is not found in the claims; from the Appeal Brief,

Appellant's argument appears to assume that the disputed claim limitation reads "wherein a first

portion of code allowing a second portion of code to access *a common resource* [i.e. /tmp/] does not

depend on the second portion of code allowing the first portion of code to access *the common*

resource".

      Independent claims 10, 12, 15, and 17 are substantially similar to claim 1 as above;

accordingly, Examiner respectfully submits that the rejections of those claims be maintained for at

least the reasons discussed above. Additionally, as Appellant's arguments regarding dependent

claims 2-9, 11, 13, 14, 16, and 18-25 stand or fall with independent claims 1, 10, 12, 15, and 17,

again Examiner respectfully requests that the rejections of those claims should also be maintained for

substantially similar reasons.

---

[1] It is noted that, as "first" and "second" are arbitrary designations vis-à-vis "portions of code", thus a.x is understood to

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related

Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

/Thomas Gyorfi/

Thomas Gyorfi

Examiner, Art Unit 2135

Conferees:

/KimYen Vu/

Supervisory Patent Examiner, Art Unit 2135

/HOSUK SONG/
Primary Examiner, Art Unit 2135